**Listing of Claims (including status and amendments):**

1      1. (Currently amended) A method of managing retrieval of messages from a shared queue, where

2      a plurality of receiver[[ing]] applications [[are]] retrieve[[ing]] messages from the same shared

3      queue, where a PUT operation by a sender application is used to place a message onto the shared

4      queue, and where a message on the queue is not to be [[accessed]] retrieved until a subsequent

5      commit of the PUT operation occurs with each message on the queue having been sent by a

6      sender application program, the method comprising:

7      assigning an index key to a message in response to [[such]] a commit of the PUT operation,

8      wherein the assigned index key comprises an attribute value of the message which was specified

9      by the sender[[ing]] application when the message was [[sent]] placed on the queue, so that the

10      assignment of the index key is deferred until a commit occurs of the PUT operation[[, whereby,

11      prior to such commit, the message will be on the queue but no index key will be assigned to the

12      message and therefore the message cannot be received by a receiver application]]; and

13      in response to a receiver application [[program]] requesting retrieval of messages from the queue

14      and specifying [[the]] a request attribute value, monitoring the availability of messages in the

15      queue with reference to said assigned index key to identify a match,

16      whereby the index key being [[so]] assigned to the message in response to said commit

17      [[provides an index which]] is usable for identifying [[only]] committed messages having an

18      attribute value matching the [[particular application-specified]] request attribute value.

- 3 -

1     2.(Currently amended)     A method according to claim 1 <u>wherein the receiver application</u>

2     <u>specifies other criteria for a match</u>, <u>said method further</u> including <u>the step of</u>:

3     in response to the monitoring step identifying the availability of a committed message in

4     the queue which has the assigned index key, determining whether the message matches other

5     criteria of the retrieval request; and

6     in response to a positive match, sending a response to the <u>receiver</u> application [[program]]

7     which issued the request.


1     3.(Currently amended)     A method according to claim 2, wherein the [[response]]

2     <u>monitoring step identifies to the requestor application the</u> [[includes]] the message <u>corresponding</u>

3     <u>to the match</u> [[which matches the request]] .


1     4.(Original)     A method according to claim 1, wherein the attribute value included in the

2     assigned index key is a message identifier or a correlation identifier.


1     5.(Currently amended)     A method according to claim 1, wherein receiver application

2     programs are able to issue retrieval requests with a wait attribute, and wherein the method

3     includes:

4     responsive to <u>identifying</u> no messages which match the request<u>ed attribute value</u> [[being

5     available in the queue when the request is issued]], triggering a monitoring process to <u>repeatedly</u>

6     perform the monitoring step <u>for the request attribute value</u>; and

7     responsive to the monitoring step identifying the availability of a committed message in

8     the queue having said assigned index key, determining whether the message matches a waiting

- 4 -

9 ~~retrieval request and, if matching, sending a response to the requestor application program which~~

10 ~~issued the request.~~

1 6. (Currently amended)      A method according to claim 1, wherein the queue is a shared

2 [[access]] queue held in a list structure of a Coupling Facility to which a plurality of resource

3 managers can connect to put messages on the queue and to retrieve messages from the <u>shared</u>

4 queue on behalf of respective sender and receiver applications [[programs]].

1 7. (Original)   A method according to claim 6, wherein the step of assigning an index key at

2 commit time comprises a resource manager which put the message on the shared queue

3 providing the attribute value to the Coupling Facility in response to committing the put

4 operation, the Coupling Facility then building the index key and storing it in association with the

5 enqueued message.

1 8.(Currently amended)      A method according to claim 6, wherein the assigned index value

2 for each message is held in a predefined control data area of the Coupling Facility list structure

3 which holds the <u>shared</u> queue.

1 9.(Original)   A method according to claim 8, wherein the predefined control data area of the

2 Coupling Facility list structure is a Coupling Facility list entry control data area, and the

3 predefined control data area holds a message identifier and a correlation identifier for the

4 message, the assigned index key comprising one of said message identifier or correlation

5 identifier.

- 5 -

1    10.(Currently amended)       A method according to claim 6, wherein the monitoring step is

2    performed by a monitoring process within the Coupling Facility in response to receipt of a

3    retrieval request which specifies [[said]] the request attribute value, the monitoring process

4    including:

5         means for determining whether an identified available message matches all criteria of the

6    received retrieval request; and

7         means, responsive to a positive match, for sending a response to the application program

8    which issued the request.


1    11.(Currently amended)       A method according to claim 10, wherein the monitoring process is

2    adapted to invoke said means for determining a match for all identified messages in said queue

3    which have said assigned index value corresponding to the request [[application-specified]]

4    attribute value.


1    12.(Cancelled) A method according to claim 1, wherein the assigned key comprises a secondary

2    index key representing a sender-application-assigned attribute and can be used to identify

3    messages in response to a retrieval request which specifies said attribute, and an additional

4    primary index key comprising sequencing information is assigned to a message when the

5    message is placed on the queue; and wherein the primary index key is used to select a message

6    for retrieval from the available messages identified in the monitoring step which used the

7    secondary key.


1    13.(Currently amended)       A program product comprising program code including executable

2    program instructions recorded on a machine-readable recording medium, for controlling the

- 6 -

3    performance of operations of a data processing apparatus on which it executes relating to storage

4    and retrieval of messages on a shared queue where a PUT operation is used to place a message on

5    the queue but access is not intended until a subsequent commit action is taken, the program code

6    including:

7              program code for postponing the assigning an index key to a message to be in response to

8    such commit of the PUT operation, wherein the assigned index key comprises an attribute value

9    of the message which was specified [[by the sending application]] when the message was sent;

10   and

11             program code, responsive to a receiver application program requesting retrieval of

12   messages from the queue and specifying [[the]] a request attribute value, for monitoring the

13   availability of messages in the queue with reference to said assigned index key, whereby the

14   index key assigned to the message in response to said commit provides an index which is usable

15   for identifying only committed messages having the [[particular application-specified]] attribute

16   value that matches the requested attribute value.


1    14.(Cancelled) A program product according to claim 13, wherein the program code includes:

2              means, responsive to the monitoring step identifying the availability of a committed

3    message in the queue which has the assigned index key, for determining whether the message

4    matches other criteria of the retrieval request; and

5              means, responsive to a positive match, for sending a response to the application program

6    which issued the request.


7    15. (Currently amended)        A resource manager component for a data processing apparatus, for

8    storing messages within a queue and storing index keys in association with the enqueued

- 7 -

9  messages for use in retrieval of the messages from the queue, the resource manager component

10  including:

11  means for assigning an index key to a message after a delay to be in response to commit

12  of the operation of putting the message on the queue, wherein the assigned index key comprises

13  an attribute value of the message which was specified by the sending application when the

14  message was sent; and

15  means, responsive to a receiver application program requesting retrieval of messages

16  from the queue and specifying [[the]] a request attribute value, for monitoring the availability of

17  messages in the queue with reference to said assigned index key, whereby the index key assigned

18  to the message in response to said commit provides an index which is usable for identifying only

19  committed messages having the particular [[application-specified]] the request attribute value.

- 8 -

1     16.(Currently amended)     A data processing apparatus including:

2       storage means;

3       a data processor;

4       a resource manager component for storing messages within a queue and storing index

5 keys in association with the enqueued messages for use in retrieval of the messages from the

6 queue, the resource manager component including:

7       means for assigning an index key to a message after a delay to be in response to commit

8 of the operation of putting the message on the queue, wherein the assigned index key comprises

9 an attribute value of the message which was specified by the sending application when the

10 message was sent; and

11       means, responsive to a receiver application program requesting retrieval of messages

12 from the queue and specifying [[the]] a request attribute value, for monitoring the availability of

13 messages in the queue with reference to said assigned index key, whereby the index key assigned

14 to the message in response to said commit provides an index which is usable for identifying only

15 committed messages having the ~~particular application-specified~~ request attribute value.


1     17. (Canceled) A data processing apparatus according to claim 16, wherein the resource manager

2 component includes means, responsive to the monitoring step identifying the availability of a

3 committed message in the queue which has the assigned index key, for determining whether the

4 message matches other criteria of the retrieval request, and the apparatus further includes means,

5 responsive to a positive match, for sending a response to the application program which issued

6 the request.

- 9 -